A Modeling Strategy for Developing Genomic Identifier CDEs

Craig Street, Rakesh Nagarajan, Vishal Nayak, and Juli Klemm

12 January 2005

A modeling strategy is presented that provides an approach for integrating all the available collections of biomedical information that will be represented as a service on the **ca**ncer **B**iomedical **I**nformatics **G**rid (caBIG) for *ad hoc* needs. This strategy is a revision of the recommendations made in the CDE white paper prepared by Rakesh Nagarajan of the Genome Annotation Special Interest Group within the Integrative Cancer Research (ICR) Workspace. The whitepaper is available at http://cabig.nci.nih.gov/workspaces/ICR/Meetings/SIGs/gene_annotation/Older_Teleconferences/Gene%20CDE%20Focus%20Group/20041007_GeneCDE_whitepaper

Recommendations

The white paper recommends using the **ca**ncer **D**ata **S**tandards **R**epository (caDSR) to store Unified Modeling Language (UML) models of interrelated objects comprising of ISO/IEC 11179 compliant **C**ommon **D**ata **E**lements (CDEs). It also recommends leveraging the controlled vocabularies provided by the **E**nterprise **V**ocabulary **S**ervices (EVS) to define the data element concepts and value domains that combine to form the CDEs. The following steps are outlined for the ICR project developers:

- Each project must describe its objects using a UML model whose classes and attributes are described by terms in the EVS.
- This model must be imported into the caDSR using the UML loader, whereby data elements are created.
- The same data elements representing genomic identifiers should be used across projects.

Using the CDEs across different projects facilitates syntactic and semantic integration of data. However, the absence of a common identifier for genes and gene products across all publicly available databases poses complications. Therefore, we make the following recommendations:

- A tentative list of required genomic identifiers (i.e., DNA or its RNA or protein product) has been compiled after polling the list of ICR projects (see Table 1).
- This list of required Genomic Identifier CDEs will be created *a priori* by members of the Genome Annotation SIG and representatives from the Architecture and Vocabulary/CDE Workspaces.
- Each ICR project's object model must utilize **at least one** of these defined genomic identifier CDEs for each UML class with any genomic identifier attributes.
- Because the goal is not to be restrictive, if object models in the future cannot reasonably accommodate one of the existing CDEs, additional CDEs may be added to this dynamic list.

Table 1 *(Still being Finalized)*

- Ensembl ID
- GenBank Accession Number

- LocusLink ID
- RefSeq mRNA Accession
- RefSeq Protein Accession
- UniGene ID
- UniProt ID

In keeping up with these recommendations, the following UML modeling approach is proposed that provides a flexible solution to deal with the problem of multiple identifiers for the same objects that is in sync with the long term proposal of providing a mapping service for the gene and gene product identifiers.

The Genomic Identifier Property

A separate identifier term for genomic classes - "Genomic Identifier" - can be defined to impart more specificity; that is, to distinguish it from other identifiers such as those for patients, tissues, etc. In defining the attributes of their genomic classes, developers are only constrained by having to reference at least one of the existing Genomic Identifier CDEs. There will be CDEs created for approved genomic identifiers of Gene, RNA and Protein *object classes* and the number of CDEs depends on the *property qualifier* (e.g., since LocusLink contains gene, RNA, and protein information, three CDEs will be created when the qualifier is "LocusLink", but only one CDE will be created when the qualifier is "RefSeq mRNA Accession"). Domain experts will ensure that the appropriate complement of classes and identifiers are created.

These CDEs can be reused across multiple applications and provide a means of interlinking the various object models. The idea is to provide flexibility to the individual developers to come up with relevant classes for their projects, interlink the appropriate class attribute with the correct genomic identifier property (thus promoting the reuse of CDEs) - thereby making them recognizable across the grid. If the approved identifier list is not sufficient, new property qualifiers may be added to the appropriate classes. If the individual developers need to define specific attributes for their classes, they may develop appropriate CDEs and add them to the caDSR repository (if not present). If the genomic classes use a common identifier between them, a join can be made by querying on any set of the four concept codes (stored in the EVS) comprising a CDE: the object class, the class qualifier, the property, and/or the property qualifier.

These objects are still extensible (because individuals either create new CDEs or map to previously existing ones). By restricting the property qualifiers (i.e., acceptable identifiers) and the corresponding value domain, the caDSR could help in validating acceptable identifiers and gene and gene product classes (by ensuring that each class uses at least one identifier CDE).

An Illustrative Example

The following example shows how one would query disparate object types mapped with the same identifier:

For example, if Georgetown and Washington University map the BRCA1 LocusLink gene:

LocusLink ID 672 BRCA1: breast cancer 1, early onset

Washington University

CDE-->DEC--->Object Class = Ribonucleic Acid (EVS concept code: C812)
CDE-->DEC--->Class Qualifier = Messenger RNA (EVS concept code: C813)
CDE-->DEC--->Property = Genomic Identifier (EVS concept code: to be created)
CDE-->DEC--->Property Qualifier = LocusLink (EVS concept code: 2184527)
CDE-->VD--->Syntax constrained to LocusLink identifiers, but not enumerated

Georgetown University

CDE-->DEC--->Object Class = Protein (EVS concept code: C17021)
CDE-->DEC--->Class Qualifier = ***optional (one could specify the protein family)***
CDE-->DEC--->Property = Genomic Identifier (EVS concept code: to be created)
CDE-->DEC--->Property Qualifier = LocusLink ID (EVS concept code: 2184527)
CDE-->VD--->Syntax constrained to LocusLink identifiers, but not enumerated

Note that one could query across LocusLink Identifier even if one maps to an RNA class and one to a protein class. The concept codes of the *property* and *property qualifier* could be used to restrict the query regardless of the *object class* or *class qualifier* concept codes.

Caveats

Note that the caDSR currently does not have a means of imposing a constraint that the biological classes should use at least one of a set of identifiers. Therefore, developers and curators will have to ensure that at least one accepted identifier is used. Representative from the Architecture and V/CDE workspaces will be charged with inspecting UML models to ensure correct modeling of genomic identifiers.

Conclusions

The modeling approach described above facilitates:

- Flexibility
- Extensibility
- Reuse of CDEs

While the caveat mentioned above places some hindrances initially in the loading of UML models developed using this approach into the caDSR, the impetus for this issue to be resolved sooner rather than later is provided by the advantages this approach offers in terms of satisfying the primary goal of interlinking various caBIG data sources and applications within the grid and the promise it offers in terms of providing a viable mapping service in the future.